AMENDMENTS TO THE CLAIMS

Please amend the claims as follows.

1. (Currently Amended) A method to trace an instrumented program, comprising:

triggering a trap instruction in the instrumented program during tracing of the instrumented program;

transferring control of the instrumented program to a trap handler associated with the trap instruction;

calling into a tracing framework, by the trap handler, to perform a tracing operation associated with the trap instruction;

performing the tracing operation to obtain tracing information, wherein the tracing information is used to analyze the instrumented program; and

emulating, after performing the tracing operation, an original instruction in the instrumented program using the trap handler,

wherein the original instruction is associated with the trap instruction, [[and]]

wherein the original instruction relates to creating or dismantling a stack frame,

wherein emulating the original instruction comprises emulating a pushl instruction, and

wherein emulating the pushl instruction comprises:

obtaining a stack pointer location, wherein the stack pointer location corresponds to a location in the stack frame;

incrementing an instruction pointer to obtain an incremented instruction pointer;

loading the incremented instruction pointer in the stack frame at one location before the stack pointer location;

loading a code segment (CS) value stored one location after the stack pointer location into the stack pointer location;

loading an EFLAGS value stored two locations after the stack pointer location into one location after the stack pointer; and

loading a base pointer into two locations after the stack pointer location.

2. (Previously Presented) The method of claim 1, further comprising:

replacing the original instruction with the trap instruction in the instrumented program.

3. (Cancelled)

4. (Cancelled)

5. (Cancelled)

6. (Currently Amended) The method of claim [[5]] 1, further comprising:

decrementing the stack pointer location by one location; and

issuing a return from interrupt instruction.

7. (Currently Amended) The method of claim [[5]] 1, wherein the instruction pointer is loaded at the stack pointer location.

8. -15. (Cancelled)

16. (Currently Amended) The method of claim 1, wherein emulating the original instruction further comprises emulating a leave instruction.

17. (Previously Presented) The method of claim 16, wherein emulating a leave instruction comprises:

obtaining a stack pointer location, wherein the stack pointer location corresponds to a first location in the stack frame;

obtaining a base pointer location, wherein the base pointer location corresponds to a second location in the stack frame;

loading a base pointer obtained at the base pointer location into a base pointer register;

loading an EFLAGS value stored two locations after the stack pointer location into the base pointer location;

loading a code segment (CS) value stored one location after the stack pointer location into one location before the base pointer location;

incrementing an instruction pointer to obtain an incremented instruction pointer;

loading the incremented instruction pointer in the stack frame at two locations before the base pointer location; and

loading the instruction pointer at three locations before the base pointer.

18. (Original) The method of claim 17, further comprising:

setting the stack pointer location to three locations before the base pointer location;

incrementing the stack pointer location by one location; and

issuing a return from interrupt instruction.

19. -36. (Cancelled)

37. (New) A method to trace an instrumented program, comprising:

triggering a trap instruction in the instrumented program during tracing of the instrumented program;

transferring control of the instrumented program to a trap handler associated with the trap instruction;

calling into a tracing framework, by the trap handler, to perform a tracing operation associated with the trap instruction;

performing the tracing operation to obtain tracing information, wherein the tracing information is used to analyze the instrumented program; and

emulating, after performing the tracing operation, an original instruction in the instrumented program using the trap handler,

wherein the original instruction is associated with the trap instruction,

wherein the original instruction relates to creating or dismantling a stack frame, and

wherein emulating the original instruction comprises emulating an enter instruction

wherein emulating the enter instruction comprises:

obtaining a stack pointer location, wherein the stack pointer location is corresponds to a location in the stack frame;

incrementing an instruction pointer to obtain an incremented instruction pointer;

loading the incremented instruction pointer in the stack frame at one location before the stack pointer location;

loading a code segment (CS) value stored one location after the stack pointer location into the stack pointer location;

loading an EFLAGS value stored two locations after the stack pointer location into one location after the stack pointer;

loading a base pointer into two locations after the stack pointer location; and

loading the base pointer into a base pointer register.

38. (New) The method of claim 37, further comprising:

decrementing the stack pointer location by one location; and

issuing a return from interrupt instruction.

39. (New) The method of claim 37, wherein the instruction pointer is loaded at the stack pointer location.

40. (New) A method to trace an instrumented program, comprising:

triggering a trap instruction in the instrumented program during tracing of the instrumented program;

transferring control of the instrumented program to a trap handler associated with the trap instruction;

calling into a tracing framework, by the trap handler, to perform a tracing operation associated with the trap instruction;

performing the tracing operation to obtain tracing information, wherein the tracing information is used to analyze the instrumented program; and

emulating, after performing the tracing operation, an original instruction in the instrumented program using the trap handler,

wherein the original instruction is associated with the trap instruction,

wherein the original instruction relates to creating or dismantling a stack frame,

wherein emulating the original instruction comprises emulating a popl instruction,

wherein emulating the popl instruction comprises:

obtaining a stack pointer location, wherein the stack pointer location corresponds to a location in the stack frame;

loading a base pointer obtained from three locations after the stack pointer location into a base pointer register;

loading an EFLAGS value stored two locations after the stack pointer location into three locations after the stack pointer location;

loading a code segment (CS) value stored one location after the stack pointer location into two locations after the stack pointer location;

incrementing an instruction pointer to obtain an incremented instruction pointer; and

loading the incremented instruction pointer in the stack frame at one location before the stack pointer location.

41. (New) The method of claim 40, further comprising:

incrementing the stack pointer location by one location; and

issuing a return from interrupt instruction.

42. (New) The method of claim 40, wherein the instruction pointer is loaded at the stack pointer location.